Дизайн 2 - Графи

Предложете колкото е възможно по-бързи (в асимптотичен смисъл) и оптимални по памет алгоритми за проблемите по-долу. Ако не е уточнено в условието считаме, че даденият граф е неориентиран.
Означения: G(V, E), i, j ∈ V – фиксирани произволни върхове.
DAG – ориентиран ацикличен граф
G^R: графът, който се получава като сменим посоката на ребрата на G.

Дефиниции:

- Pre-order number, post-order number: номерата, които върховете получават при обхождане с DFS съответно преди и след обхождане на техните наследници.
- Диаметър: най-дългият измежду най-късите пътища между всеки два върха.
- Срязващ връх (cut vertex, articulation point): връх, който ако бъде изтрит заедно с инцидентните с него ребра, ще се увеличи броя на свързаните компоненти на графа.
- Мост (bridge): ребро, което ако бъде изтрито, ще се увеличи броя на свързаните компоненти на графа.
- Клика: пълен подграф. Кликово число на граф: броят на върховете в максимална по мощност клика. Антиклика (независимо множество): празен подграф (без съседни върхове). Число на независимост на граф: броят на върховете в максимална по мощност антиклика.
- Доминиращо множество: подмножество D на V, такова че всеки връх, който не принадлежи на D е пряк наследник на някой връх от D. Число на доминиране на граф: броят на върховете в минимално по мощност доминиращо множество.
- Връх i в ориентиран граф. Полустепен на входа i: броя на влизащите ребра; полустепен на изхода на i: броя на влизащите ребра. Sink: връх, от който не излизат ребра. Source: връх, към който не влизат ребра.
- Свързан ориентиран граф: за всеки два върха i и j има път от i до j или от j до i. Силно свързан: за всеки два върха i и j има път от i до j и от j до i. Силно свързана компонента (SCC): максимален подграф, в който всеки два върха са силно свързани.
- Нека силно свързаните компоненти на ориентиран граф са C_1, C_2, … $C_к$. Компонентен граф: граф с върхове { C_1, C_2, … $C_к$ } и ребра: има ориентирано ребро от C_l до C_t, ако съществуват върхове i в C_l и j в C_t: има път от i до j в G.
- Хамилтонов цикъл: минава през всеки връх точно по веднъж.
- Нека G е свързан мултиграф. Ойлеров цикъл - в който всяко ребро участва точно по веднъж. Аналог. Ойлеров път.
- T – покриващо дърво на G. Bottleneck edge: най-тежкото ребро в T. MBST(min bottleneck spanning tree): покриващо дърво с минимална стойност на bottleneck edge; bottleneck: тежестта на това ребро.

Основни задачи

Да се докаже, че:

1. Неориентиран граф с n върха и поне n ребра е цикличен.
2. В DAG има поне един sink и поне един source.

Да се намери:

1. Броят на свързаните компоненти на граф.
2. Дали граф е цикличен.
3. Най-късият път (по брой ребра) от връх i до връх j.
4. A) Най-късият път в претеглен граф с неотрицателни ребра от връх i до връх j.
   B) -//- ако има отрицателни ребра.
5. Дължините на най-късите пътища в претеглен граф между всеки два върха.
6. Броят на пътищата от връх i до връх j с дължина k ребра.
7. Дадени са градове и цените за построяване на велоалеи между някои от тях. Да се намери каква е минималната сума, с която може да се построят велоалеи, така че да има път между всеки два града.
8. A) Дадени са задачи и зависимости между тях от типа: задача i трябва да се изпълни преди задача j. Да се намери ред за изпълнение на задачите (ако съществува).
   B) Да се подредят върховете на DAG така че всеки връх да се намира след своите наследници.
9. Най-близкият общ предшественик на два върха в дърво (възможно най-отдалечения от корена).

Други задачи

1. Всички прости пътища от връх i до връх j.
2. Дали граф е Хамилтонов. Хамилтоновия цикъл с минимална дължина.
3. A) Броят на простите пътища от връх i до връх j в DAG.
   B) Дължината на най-дългия път в DAG.
4. Диаметърът на дърво.
5. A) Броят на срязващите върхове.
   B) Броят на мостовете.
6. A) Кликовото число и числото на независимост на дърво.
   B) -//- на граф. Всички максимални по включване антиклики в граф.
7. Числото на доминиране на граф.
8. A) Неориентиран свързан мултиграф. Да се докаже, че съдържа:
   - Ойлеров цикъл ⇔ всички върхове са от четна степен.
   - Ойлеров път ⇔ има точно два върха от нечетна степен.
   B) Ориентиран свързан мултиграф. Да се докаже, че съдържа:
   - Ойлеров цикъл ⇔ за всеки връх полустепените на входа и на изхода са равни.

А за Ойлеров път?
C) Да се намери Ойлеров цикъл. А път?
9. A) Да се докаже, че компонентният граф е DAG.
B) Нека C е SCC в G, i връх в C. Да се докаже, че множеството от достижими от i върхове в G съвпада с C.
C) Да се докаже, че върхът с най-голям post-order number принадлежи на source SCC в G.
D) Да се намерят силно свързаните компоненти на граф.
10. Ребро, което участва във всички цикли в ориентиран граф (ако съществува).
11. Покриващо дърво, което минимизира:
A) най-лекото ребро
B) най-тежкото ребро (MBST).
C) Да се провери дали bottleneck-а е число по-малко или равно на c (c = const).
12. Второто по оптималност минимално покриващо дърво.
13. Частично покриващо дърво на G с k (k<=n) върха: покриващо дърво на графа G_1, състоящ се от тези k върха и инцидентните им ребра от G. Минималното измежду всички частични покриващи дървета с произволни k върха на G?